DECIDE FOR

```
DECIDE FOR {FIRST} CONDITION
           {EVERY}

        {WHEN logical-condition statement..}...

        [WHEN ANY statement...]

        [WHEN ALL statement...]

        WHEN NONE statement...

END-DECIDE
```

Related Statements: DECIDE ON | IF

Function

The DECIDE FOR statement is used to decide for one or more actions depending on multiple conditions (cases).

Note:
If **no** action is to be performed under a certain condition, you specify the statement IGNORE in the corresponding clause of the DECIDE FOR statement.

FIRST/EVERY

With the keyword FIRST or EVERY, you indicate whether only the first or every true condition is to be processed.

WHEN logical-condition

With this clause, you specify the *logical condition(s)* to be processed. See the section Logical Condition Criteria in the Natural Reference documentation.

WHEN ANY

With WHEN ANY, you can specify the *statement(s)* to be executed when *any* of the logical conditions are true.

WHEN ALL

With WHEN ALL, you can specify the *statement(s)* to be executed when *all* logical conditions are true. This clause is applicable only if EVERY has been specified.

WHEN NONE

With WHEN NONE, you specify the *statement(s)* to be executed when *none* of the logical conditions are true.

Example 1

```
   /* EXAMPLE 'DECEX1:' DECIDE FOR (USING FIRST OPTION)
   /***************************************************
   /* IF FUNCTION = A AND PARM = X
   /*    ROUTINE-A IS TO BE EXECUTED.
   /* IF FUNCTION = B AND PARM = X
   /*    ROUTINE-B IS TO BE EXECUTED.
   /* IF FUNCTION = C THRU D
   /*    ROUTINE-CD IS TO BE EXECUTED.
   /* FOR ALL OTHER CASES,
   /*    REINPUT STATEMENT IS TO BE EXECUTED.
   /***************************************************
   DEFINE DATA LOCAL
   1 #FUNCTION (A1)
   1 #PARM (A1)
   END-DEFINE
   /***************************************************
   INPUT #FUNCTION #PARM
   /***************************************************
   DECIDE FOR FIRST CONDITION
              WHEN #FUNCTION = 'A' AND #PARM = 'X'
                PERFORM ROUTINE-A
              WHEN #FUNCTION = 'B' AND #PARM = 'X'
                PERFORM ROUTINE-B
              WHEN #FUNCTION = 'C' THRU 'D'
                PERFORM ROUTINE-CD
              WHEN NONE
                REINPUT 'PLEASE ENTER A VALID FUNCTION'
                         MARK *#FUNCTION
   END-DECIDE
   /***************************************************
   END
```

```
   #FUNCTION A #PARM Y
```

```
   PLEASE ENTER A VALID FUNCTION
   #FUNCTION A #PARM Y
```

Example 2   DECIDE FOR

Example 2

```
   /* EXAMPLE 'DECEX1E:' DECIDE FOR (EVERY OPTION)
   /****************************************************************
   DEFINE DATA LOCAL
   1 #FIELD1 (N5.4)
   END-DEFINE
   /****************************************************************
   INPUT #FIELD1
   /****************************************************************
   DECIDE FOR EVERY CONDITION
           WHEN #FIELD1 >= 0
             WRITE '#FIELD1 is positive or zero.'
           WHEN #FIELD1 <= 0
             WRITE '#FIELD1 is negative or zero.'
           WHEN FRAC(#FIELD1) = 0
             WRITE '#FIELD1 has no decimal digits.'
           WHEN ANY
             WRITE 'Any of the above conditions is true.'
           WHEN ALL
             WRITE '#FIELD1 is zero.'
           WHEN NONE
             IGNORE
   END-DECIDE
   /****************************************************************
   END
```

```
#FIELD1 42
```

```
Page       1                                      90-10-29  12:24:33

  #FIELD1 is positive or zero.
  #FIELD1 has no decimal digits.
  Any of the above conditions is true.
```